

### **REMARKS**

Claims 1-32 are pending in the application. Claims 6, 9-15, 21 and 24-30 have been indicated as containing allowable subject matter. Claims 1, 10-16 and 25-32 have been amended. Claims 6, 9, 21 and 24 have been canceled. Claims 33-36 have been added. Claims 1-5, 7, 8, 16-20, 22, 23, 31 and 32 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Cabrera (5,978,815) in view of Shaath et al. (6,654,864). In view of the following remarks, reconsideration and withdrawal of these grounds of rejection is requested.

### **Examiner Interview**

The Applicants thank Examiner Jenise Jackson for the courtesy and the comments offered during the Telephone Interview conducted on April 7, 2005. During the Interview, the Applicants' representative (Andrew A. Noble) and the Examiner discussed the various rejections under 35 U.S.C. § 103. Accordingly, the present Amendment has been filed.

### **Claim Rejections Under 35 U.S.C. § 103**

Claims 1-5, 7, 8, 16-20, 22, 23, 31 and 32 stand rejected under 35 U.S.C. § 103(a) as being anticipated by Cabrera et al. (U.S. Pat. No. 5,978,815) in view of Shaath et al. (U.S. Pat. No. 6,654,864). In view of the following remarks, reconsideration and withdrawal of this ground of rejection is respectfully requested. The Applicants' claim 1 has been amended.

Claim 1 now recites:

A method for providing data security in a first device driver  
operably installed in a computer operating system having a layered

plurality of device drivers for accessing data in a data storage device, the method comprising the steps of:  
detecting an I/O request to said first device driver;  
determining whether said first device driver is functionally uppermost in the layered plurality of device drivers;  
if said first device driver is functionally uppermost in the layered plurality of device drivers, performing the I/O request in said first device driver; and  
if said first device driver is not functionally uppermost in the layered plurality of device drivers, denying the I/O request in said first device driver, and allowing the I/O request to be performed by a next lower level device driver in the layered plurality of device drivers; wherein denying the I/O request in said first device driver includes implementing at least one data security measure before allowing the I/O request to be performed by the next lower level device driver. (emphasis added).

Thus, the Applicants' claim 1 requires a method for providing data security including the steps of receiving an I/O request from a calling device, (i.e., from another application such as an operating system or a device driver), "determining whether [a] first device driver is functionally uppermost in [a] layered plurality of device drivers," and if the first device driver is not functionally uppermost in the layered plurality of device drivers, performing the steps of "denying [an] I/O request in [a] first device driver" and "allowing the I/O request to be performed by a next lower level device driver." The step of denying the I/O request includes implementing at least one data security measure before allowing the I/O request to be performed by the next lower level device driver. For example, the implementation of at least one data security measure may include incrementing a counter for monitoring the number of times an I/O request is denied in the first device driver, determining whether a stored count has reached a maximum threshold and/or initiating a programmable security response. (See, pg. 12, lines 19-22, pg. 13, line 1). As explained below, Cabrera in view of Shaath et al. fail to disclose, teach or suggest such an invention.

Cabrera teaches a Microsoft Windows NT® operating system including a user mode and a kernel mode. (See, col. 6, lines 60-62). The user mode includes at least one client process 94 which makes I/O requests 98 upon an I/O manager 110 in the kernel mode. The I/O manager 110 forwards each I/O request received from the client process 96 to the layer 1 driver 100. (See, col. 16, lines 34-39). The I/O request is forwarded through various additional 'layers' (e.g., layer 2 driver, layer 3 driver,...layer N driver) with each driver performing any required processing before forwarding the I/O request to the next driver. (See, col. 16, lines 54-61).

Shaath et al. teaches a method of restricting file access to prevent certain operations in a Microsoft Windows NT® operating system environment wherein a set of file write access commands are determined from data stored within a storage medium. The set of file write access commands are for the entire storage medium. Any I/O request that matches a file write access command provided to the file system for that storage medium results in an error message. Other I/O requests are passed on to a device driver for the storage medium and are implemented. For example, one implementation may include a device driver calling I/O support routines to tell a Windows NT® I/O manager that a device operation is pending on the I/O request and to either queue or pass the I/O request to another driver supplied routine. (See, col. 6, lines 23-27). In this way, commands such as "file delete" and "file overwrite" can be disabled for an entire storage medium.

Cabrera does not disclose, teach or suggest implementing at least one data security measure at a device driver in a stack before allowing an I/O request to be performed by another device driver. Cabrera teaches a device driver that performs "any required" processing before forwarding the I/O request to the next device driver in the stack. (See, col. 16, lines 54-61).

However, if the I/O request is not intended to be received by a particular device driver, the device driver may not be required to perform any processing at all before forwarding the I/O request to the next device driver. This is not what is disclosed in the Applicants' claim 1. In contrast to Cabrera, claim 1 discloses implementing at least one data security measure at the receiving device driver before allowing another device driver to perform an I/O request. As such, data security against an unauthorized device object is provided at any device driver in the stack that receives an I/O request.

In further contrast to the Applicants' claim 1, Cabrera does not detect the position of a device driver relative to other device drivers in a stack. In Cabrera, an I/O request is merely passed to a first driver means for performing I/O processing. (See, col. 7, lines 33-35). As such, Cabrera does not disclose, teach or suggest determining whether a first device driver is functionally uppermost in a layered plurality of device drivers and, in particular, for utilizing such a determination for determining whether at least one data security measure should be implemented. Basically, an I/O request is passed from device driver 1 to device driver 2 to device driver 3, etc. for every I/O request passed to the first device driver. (See, col. 17, lines 34-36). The passing of an I/O request to a device driver, processing the I/O request and sending the I/O request "on to the next driver" suggests a sequential process rather than a determinative understanding of the relative positions of the device drivers in a stack in reference to each other. In contrast, as to Applicant's claim 1, the stack position of the particular device driver that receives an I/O request is determinative for implementing various data security measures and at least one data security measure will be implemented before allowing the I/O request to be performed by another device driver in the stack.

In addition, it would not be obvious to combine Cabrera with the security measures disclosed in Shaath et al. Cabrera's only reference to security is that security may be provided by an operating system that may work in conjunction with the invention. (See, col. 7, lines 7-13). Further, Cabrera teaches away from providing data security as contemplated in Applicants' claim 1 by noting that security is less of a priority at the kernel (i.e., device driver) level. (See, col. 7, lines 23-27). However, even if, arguendo, Cabrera did contemplate providing data security measures at the device operation level and could be properly combined with Shaath et al. as the Examiner suggests, Shaath et al. does not disclose, teach or suggest implementing security measures when a first device driver receiving an I/O request packet is not functionally uppermost in the layered plurality of device drivers. For example, in Shaath et al. a scenario can be envisioned in which an unauthorized device driver may include a set of file write access commands for an entire storage medium and interpose itself within the device driver stack. As such, any I/O request received by the unauthorized device driver would thus be processed by the unauthorized device driver, possibly resulting in a security breach. While the method in Shaath et al. contemplates preventing certain write access commands sent via an I/O request, it does not prevent against the imposition of an unauthorized device driver or application. Therefore, the security measures disclosed in Shaath et al., if they can even be contemplated in light of Cabrera, are not adequate for detecting when another application attempts to obtain priority for receiving an I/O request over a first device driver, e.g., a file system monitor, such as by interposing itself between the file system monitor and the calling device, e.g., a Windows NT® I/O manager.

Accordingly, reconsideration and withdrawal of this rejection with respect to claims 1-5, 7 and 8, is respectfully requested. Independent claims 16, 31 and 32 have been amended

similarly to claim 1. As such, for at least the reasons cited for claim 1 above, reconsideration and withdrawal of this rejection with respect to claims 16, 31 and 32 and claims 17-20, 22 and 23, which depend from claim 16, is respectfully requested.


The Applicants would like to thank the Examiner for indicating that claims 6, 9, 21 and 24 contain allowable subject matter. As such, new independent claims 33-36 include the limitations of canceled claims 6, 9, 21 and 24 respectively and should now therefore be in a condition for allowance. Favorable consideration of these claims is requested.

Therefore, for those reasons discussed above, reconsideration and withdrawal of this ground of rejection is respectfully requested.

### **Conclusion**

In view of the foregoing remarks, Applicants submit that this application is in condition for allowance at an early date, which action is earnestly solicited.

Respectfully submitted,



---

Paul A. Taufer  
Reg. No. 35,703  
Andrew A. Noble  
Reg. No. 48,651

DLA Piper Rudnick Gray Cary US LLP  
One Liberty Place  
1650 Market Street, Suite 4900  
Philadelphia, PA. 19103  
Phone: 215.656.3300